Advances in Computing 2017, 7(2): 51-57
DOI: 10.5923/j.ac.20170702.09

Improving TORA Protocol using Ant Colony Optimization Algorithm

Merril Roshini Mathias, Rayan Rahul D'Souza, Sanjana Bhat H., Savina Jassica Colaco, Renuka Tantry*

Department of Computer Science and Engineering, St Joseph Engineering College, Mangaluru, India

Abstract A wireless ad hoc network (WANET) or MANET is a decentralized type of wireless network and it does not rely on a pre-existing infrastructure. Instead, each node participates in routing by forwarding data for other nodes, so the determination of which nodes forward data is made dynamically on the basis of network connectivity and the routing algorithm in use. The Temporally-Ordered Routing Algorithm (TORA) is an adaptive, distributed, loop-free routing protocol for multi-hop networks which has minimum overhead against topological changes. Quality of Service (QoS) support for MANET in TORA has become a challenging task due to its dynamic topology. This paper proposes QoS enabled Temporally Ordered Routing Algorithm using Ant Colony Optimization called AntTORA. ACO algorithms have shown to be a good technique for developing routing algorithms for ad hoc networks. ACO based routing is an efficient routing scheme based on the behaviour of foraging ants. The collective behaviour of ants helps to find the shortest path from the nest to a food source, by deposition of a chemical substance called pheromone on the visited nodes. ACO technique is used in TORA protocol to optimize multiple QoS metrics like end-to-end delay, throughput, jitter and so on. The performance of TORA and AntTORA are analysed using network simulator-2. The results presented in the end also help the researchers to understand the differences between TORA and AntTORA, therefore to choose appropriate protocol for their research work. Our simulation study shows how this approach has significantly improved the performance of the ad hoc networks.

Keywords MANETs, ACO, Routing protocols, TORA, Ant-TORA

1. Introduction

A rapid growth and research has been seen in the field of Mobile Ad Hoc Networks (MANETs) due to their dynamic nature and infrastructure-less end-to-end communication. A Mobile Ad Hoc Network (MANET) is a collection of autonomous mobile nodes which dynamically form ad hoc network without the use of any infrastructure or centralized administration. It is sometimes also called as a multi-hop wireless network [1]. Routing in MANETs is a challenging task and has gained a remarkable attention from researchers worldwide. This has led to the development of different routing protocols with each protocol providing an improved version in respect to various strategies for a particular network scenario. None of the existing protocols is the best that justifies the characteristics and is suitable to perform an efficient routing. Researchers strive to uncover the efficiency of existing routing protocols by enhancing their performance.

Lot of techniques exist which can be applied on the existing protocols to make them more efficient and less vulnerable to outer attacks. In this section, we have discussed about TORA protocol. Also we have discussed the Ant Colony Optimization technique which is implemented on existing TORA routing protocol as AntTORA to enhance their performance.

2. TORA Routing Protocol

The Temporally Ordered Routing Algorithm (TORA) is an algorithm for routing data across Wireless Mesh Networks or Mobile ad hoc network. The TORA does not use a shortest path solution, an approach which is unusual for routing algorithms of this type. TORA builds and maintains a Directed Acyclic Graph (DAG) rooted at a destination. No two nodes may have the same height. The protocol performs three basic functions: rote creation, route maintenance and route erasure. Nodes need to maintain the routing information about adjacent (one hop) nodes. During the route creation and maintenance phase, nodes use a height matrix to establish a Directed Acyclic Graph (DAG) rooted at destination. The links are assigned based on the relative height metric of neighbouring nodes. During the time of mobility the DAG is broken and the route maintenance unit comes into picture to re-establish a DAG routed at the

* Corresponding author:
renukat@sjec.ac.in (Renuka Tantry)

Published online at http://journal.sapub.org/ac

destination. During rote erasure phase, a broadcast clear packet (CLR) is flooded throughout the network to erase invalid route.

The mobility and node failure in MANET dramatically increases. This lowers the performance and increases the end to end delay in transmission of data. Furthermore, it becomes more expensive. ACO uses set of distributed software agents called artificial ants which search for good solutions to a given optimization problem.

3. ANT Colony Optimization

ACO, a famous swarm intelligence approach, has taken the inspiration from real ants which are wandering around their nests to forage for search of food [2]. Upon finding food they will return back to their nests and simultaneously deposit pheromone trails along the paths. The ant selects its next hop based on the amount of pheromone deposited on the path to the next node. The problem of finding shortest paths maps quite well to the problem of routing in networks. The ants are nothing but small control packets, which have the task to find a path towards their destination and gather information about it. Because of its robustness, and adaptive nature, ACO can find its applications in routing, assignment & scheduling [3]. The network is ad hoc because it does not rely on a pre-existing infrastructure, such as routers in wired networks or access points in managed (infrastructure) wireless networks. Wireless mobile ad hoc networks are self-configuring, dynamic networks in which nodes are free to move. There are lots of similarities between mobile ad hoc networks and ants shown in the Table 1. Ant based routing algorithms exhibit a number of desirable properties for MANET routing: they work in a distributed way, are highly adaptive, robust, and provide automatic load balancing.

Table 1. Comparison between Ad hoc Networks and Ants

Parameters	MANETs	Ants
Physical Structure	Unstructured, dynamic & distributed	-do-
Origin of Route	Route Requests are sent from source to get local information	Pheromones are used to build new routes
Multipath Support	Single path, partially Multipath	Provide multipath
Basic System	Self-configuring & self-Organizing	-do-
Goal	To find the shortest Path	Guaranteed shortest path

ACO uses set of distributed software agents called artificial ants which search for good solutions to a given optimization problem. The artificial ants incrementally build solutions by moving in the network. These ants deposit pheromones in order to mark some favourable path that should be followed by other members of the colony.

Ant Colony Optimization technique is very much suitable for mobile ad hoc networks because of following characteristics:

- No single point of failure as the algorithm is fully distributive in nature
- Simple routing operations
- Ant agents fully cooperate and obey the routing rules
- No need of defining path recovery procedures as it is self organizing and fault tolerant
- Very much adaptable to the network traffic
- Compliant to the dynamically changing topology of ad hoc networks

4. AntTORA

Temporally Ordered Routing Algorithm (TORA) is a source initiated on-demand routing protocol which uses a link reversal algorithm and provides loop-free multipath routes to a destination node [4]. Each node maintains its one-hop local topology information and also has the capability to detect partitions.

TORA has the unique property of limiting the control packets to a small region during the reconfiguration process initiated by a path break. TORA uses three different control message types, Query (QRY) for route discovery, Update (UPD) to update the routing structure and height tables and Clear (CLR) to delete invalid routes.

Forward ant (FANT) and backward ant (BANT) packets are used in the route discovery process. A Forward ant is an agent which establishes the pheromone track to the source node. It gathers information about the state of network. The Forward ant is a small packet with a unique sequence number. Nodes are able to distinguish duplicate packets on the basis of the sequence number and the source address of the Forward ant. It creates a set of routing agents called Forward ant to search for the destination host. Backward ants serve the purpose to TORA and are used in route creation process. The packet formats of TORA and AntTORA are shown from figures 1 to 4.

Version #	Type	Reserved	Destination IP Address

Figure 1. TORA QRY Packet Format

Version #	Type	Reserved	Destination IP Address	QoSMetrices Seq No. Source. Destination

Figure 2. AntTORA FANT Packet Format

Version #	Type	Reserved	Destination IP Address	Model #	H.tau, H.oid, H.r, H.delta H.id

Figure 3. TORA UPD Packet Format

Version #	Type	Reserved	Destination IP Address	Model #	QoS Metrices H.tau, H.oid, H.r, H.delta H.id

Figure 4. AntTORA BANT Packet Format

5. Architectural Design

Architectural Design is an innovative idea where we try to establish a system organization that will satisfy the functional and non-functional requirements of the system.

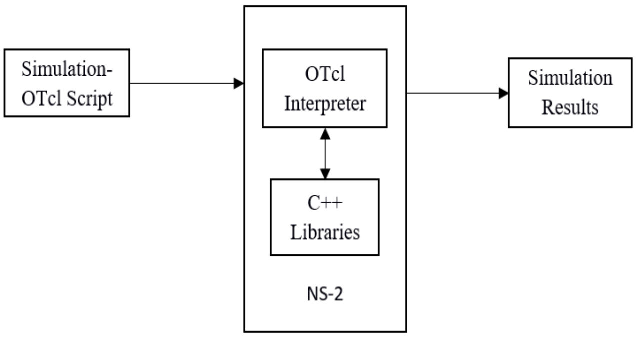


Figure 5. Architectural Diagram

In NS2 simulation, the user input is an OTcl source file, the OTcl script does the work of initiating an event scheduler, setting up the network topology using the network objects and the plumbing functions in the library, and tells traffic sources when to start and stop transmitting packets through the event scheduler. And then, this OTcl script file will be passed to NS-2.

In this view, we can treat NS-2 as Object-Oriented Tcl (OTcl) script interpreter that has a simulation event scheduler, network component object libraries and network setup module libraries. Then the detailed network construction and traffic simulation will be actually done in NS-2.

After the simulation is finished, NS produces on or more text-based output files that contain detailed simulation data, and the data can be used for simulation analysis. From the NS-2 developer view the event schedulers and most of the network components are implemented in C++ and are available to Tcl Script. Thus the lowest level of NS-2 is implemented by C++ and the Tcl Script.

6. Methodology

6.1. Basic Ant Algorithm

The basic idea of the ant colony optimization Meta heuristic is taken from the food searching behaviour of real ants. Figure 6 shows a scenario with two routes from the nest to the food place. At the intersection, the first ants randomly select the next branch. Since the route below is shorter than the upper one, the ants that take this path will reach the food place first.

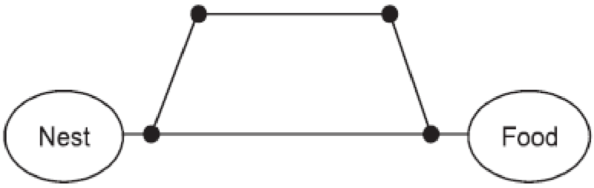


Figure 6. Ants take the shortest path after an initial searching time

On their way back to the nest, the ants again have to select a path. After a short time the pheromone concentration on the shorter path will be higher than on the longer path, because the ants using the shorter path will increase the pheromone concentration faster. The shortest path will thus be identified and eventually all ants will only use this one. This behaviour of the ants can be used to find the shortest path in networks. Especially, the dynamic component of this method allows a high adaptation to changes in mobile ad hoc network topology, since in these networks the existence of links are not guaranteed and link changes occur very often.

6.2. AntTORA

TORA protocol is an on-demand routing protocol that is based on the idea of source routing. Mobile nodes are required to maintain route caches that contain the source routes of which the mobile is aware. Entries in the route cache are continually updated as new routes are learnt. The protocol consists of two major phases: route discovery and route maintenance. Route Discovery consists of two parts: Route request and Route reply.

Route Request

When a mobile node has a packet to send to the destination, it first consults its route cache to determine whether it previously has a route to the destination. If it has an unexpired route to the destination, it will use this route to send the packet. On the other hand, if the node does not have such a route, it initiates route discovery by broadcasting a route request packet. This route request contains the address of the destination, along with the source node's address and a unique identification number. Each node receiving the packet checks whether it knows of a route to the destination. If it does not, it adds its own address to the route record of the packet and then forwards the packet along its outgoing links. To limit the number of route requests propagated on the outgoing links of a node, a mobile only forwards the route request if the request has not yet been seen by the mobile and if the mobile's address does not already appear in the route record.

Route Reply

A route reply is generated when the route request either reaches the destination itself, or reaches an intermediate node which contains in its route cache an unexpired route to the destination. By the time the packet reaches either the destination or such an intermediate node, it contains a route record yielding the sequence of hops taken. If the node generating the route reply is the destination, it places the route record contained in the route request into the route reply. If the responding node is an intermediate node, it will append its cached route to the route record and then generate the route reply. To return the route reply, the responding node must have a route to the initiator. If it has a route to the initiator in its route cache, it may use that route. Otherwise, if symmetric links are supported, the node may reverse the route in the route record. If symmetric links are not supported, the node may initiate its own route discovery and piggyback the route reply on the new route request.

Route Maintenance

Route maintenance is accomplished through the use of route error packets and acknowledgments. Route error packets are generated at a node when the data link layer encounters a transmission problem. When a route error packet is received, the hop in error is removed from the node's route cache and all routes containing the hop are truncated at that point. In addition to route error messages, acknowledgments are used to verify the correct operation of the route links. Such acknowledgments include passive acknowledgments, where a mobile is able to hear the next hop forwarding the packet along the route. In AntTORA the Forward ant (FANT) and backward ant (BANT) packets are added in the route request and route reply of TORA.

Forward ants are used to explore new paths in the network. Ants measure the current network state for instance by trip times, hop count or Euclidean distance travelled. Backward ants serve the purpose of informing the originating node about the information collected by the forward ant.

The ant routing has two types of feedback: positive feedback increases the pheromone levels on routes actively carrying ant packets and negative feedback periodically decreases pheromone values to limit the effects of stale information. Routing decisions tend to support paths with higher pheromone levels and, when allowed to converge, shortest end-to-end paths are empirically observed to be favoured. Modified ant mechanism algorithm that uses energy, delay and jitter metrics to perform updates of pheromone levels is proposed. Assuming a control packet containing energy, delay and jitter metrics, a separate pheromone level will be maintained for each metric.

In the algorithm, ant packet headers have fields that track the minimum residual energy of the nodes that relay them and track the cumulative delay and jitter based on backlog information of queued packets destined to the packet’s source. Thus, energy, delay and jitter pheromone levels will be maintained at each node.

7. Implementation of Proposed Protocol

An enhancement of the existing TORA protocol has been proposed. Ant agents have been applied while generating traffic. Ant Colony Optimization (ACO) has been implemented for computing shorter paths to the destination node based upon the pheromone trail values calculated for every valid path [5]. This value is used by a node to calculate probability of each next node which is to be next hop in order to reach the destination. The proposed work is an implementation of this collective behaviour that has been applied to ad hoc network routing protocol. Since a routing protocol completely relies on a certain set of governing rules that help in monitoring the whole communication process within an ad hoc network, hence an important role has been played by the mobile nodes in managing these processes and network resources.

8. Performance Evaluation

A rectangular grid of 800 m x 800 m has been chosen. The mobility model uses the random waypoint model in the rectangular field. The simulations have been run multiple times for variable pause times. After pausing for a specific pause time the mobile node again selects a new destination and proceeds at a speed distributed uniformly between 0 and a maximum speed.

Table 2. Simulation parameters

Simulation Parameters	
Simulation area (Grid size)	800m x 800m
Number of nodes	6
Node communication range	50 m
Protocol	TORA and AntTORA
Medium access mechanism	IEEE 802.11b
Traffic source model	Constant bit rate
Packet size	1024 Bytes
Mobility model	Random waypoint

The simulations are based on Continuous Bit Rate (CBR) sources which have been chosen to test the routing protocols. The source and destination nodes have been chosen randomly with uniform probabilities. Different parameters have been considered for simulations performed as shown in Table 2. From the available set of performance metrics four have been considered in this study and simulations have been performed on all the protocols to calculate the effective values for metrics in each case.

- *Packet Delivery Fraction:* It is the ratio of the amount of data packets delivered to the destination and total number of data packets sent by source.
- *Average End-to-End Delay:* The interval time between sending by the source node and receiving by the destination node, which includes the processing time and queuing time.
- *Throughput:* It is the average number of messages successfully delivered per unit time i.e. average number of bits delivered per second. Also refers to the amount of data transfer from source mode to destination in a specified amount of time.

9. Comparison and Result

In this section we have compared the metrics, Packet Delivery Fraction (PDF), Average En-to-End Delay and Throughput with TORA and AntTORA.

The following figure 7 tells us about the Packet Delivery Fraction in case of TORA.

The red line indicates the drop rate of packets whereas the green line indicates the packets that are received. Hence in case of TORA, the rate of packet drop is more compared to packets received. The Figure 8 shows us the Packet Delivery Fraction in case of AntTORA.

The above graph shows that the red line indicating the packet drop is very less compared to the TORA graph. But the green line indicating the packets received is high when compared to the previous graph. This indicates that Packet Delivery Fraction in case AntTORA is much better that that of TORA.

The Figure 9 displays the throughput and end-to-end results when TORA protocol is used and Figure 10 displays the throughput and end-to-end results when AntTORA protocol is used. As we can see in case of TORA, the throughput is high and end-to-end delay is also high. But there is a lot of packet drop. Hence less number of packets reach the destination. In case of AntTORA, the throughput is better even though its showing less value because out of the number of packets sent, the number of packets received is very high when compared to TORA.

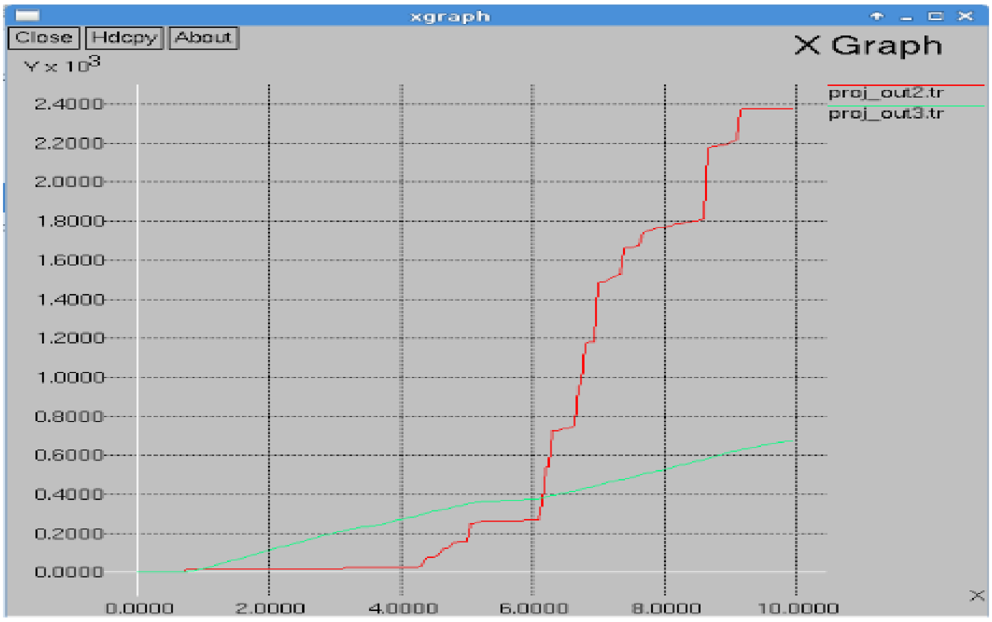


Figure 7. Packet Delivery Fraction in TORA

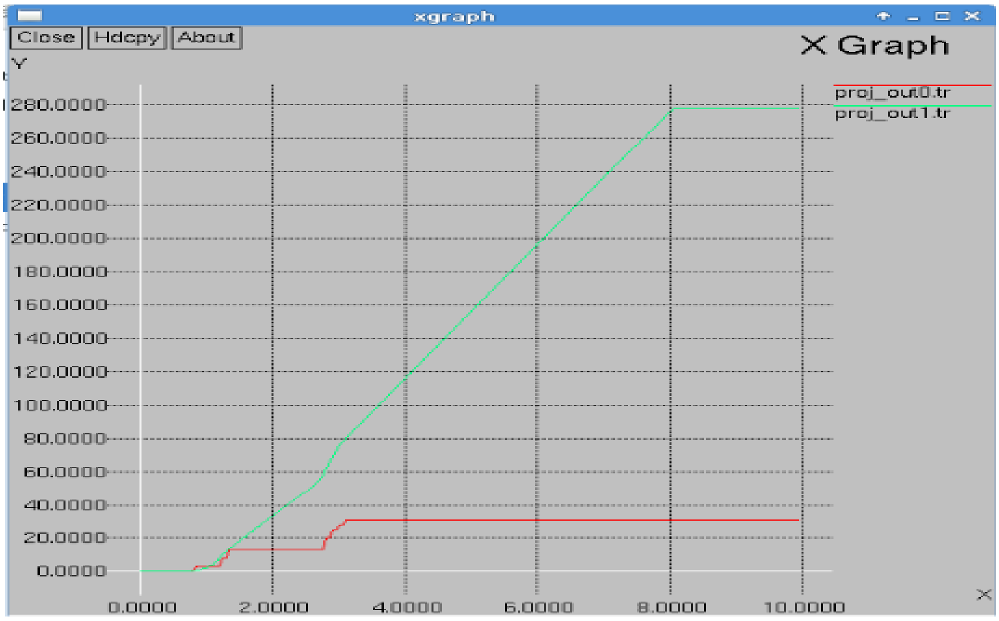
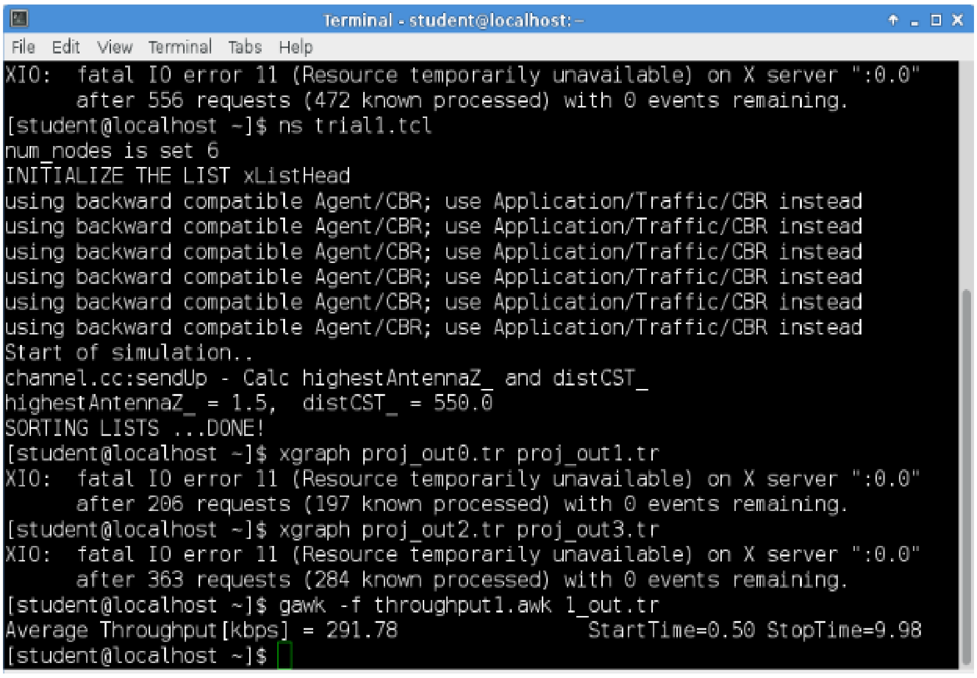
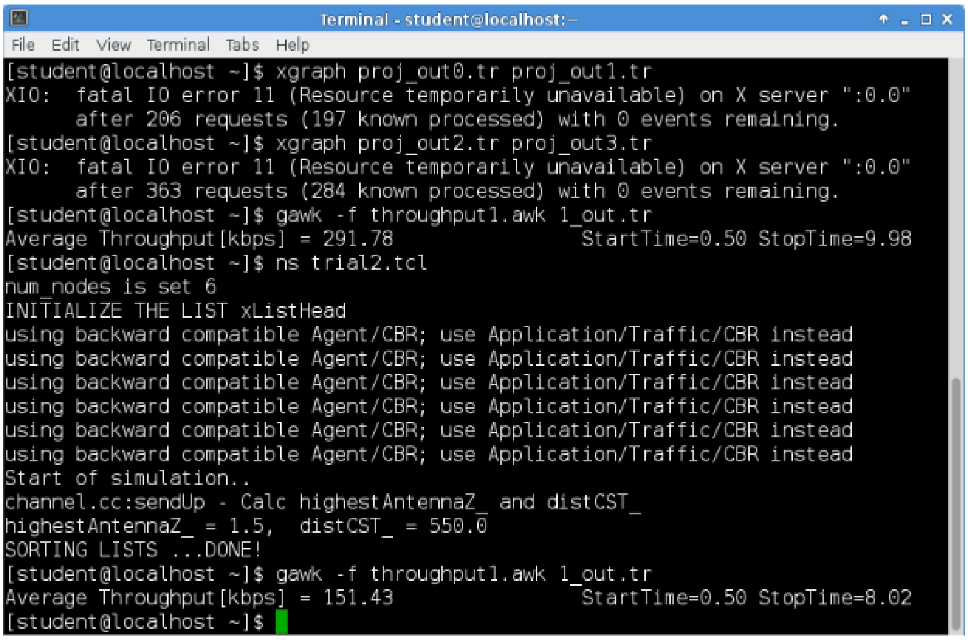


Figure 8. Packet Delivery Fraction in AntTORA


Terminal - student@localhost:~
File Edit View Terminal Tabs Help

```
XIO:  fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
      after 556 requests (472 known processed) with 0 events remaining.
[student@localhost ~]$ ns trial1.tcl
num_nodes is set 6
INITIALIZE THE LIST xListHead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
Start of simulation..
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
[student@localhost ~]$ xgraph proj_out0.tr proj_out1.tr
XIO:  fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
      after 206 requests (197 known processed) with 0 events remaining.
[student@localhost ~]$ xgraph proj_out2.tr proj_out3.tr
XIO:  fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
      after 363 requests (284 known processed) with 0 events remaining.
[student@localhost ~]$ gawk -f throughput1.awk 1_out.tr
Average Throughput[kbps] = 291.78                StartTime=0.50 StopTime=9.98
[student@localhost ~]$
```

Figure 9. Throughput and End-to-End Delay in TORA


Terminal - student@localhost:~
File Edit View Terminal Tabs Help

```
[student@localhost ~]$ xgraph proj_out0.tr proj_out1.tr
XIO:  fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
      after 206 requests (197 known processed) with 0 events remaining.
[student@localhost ~]$ xgraph proj_out2.tr proj_out3.tr
XIO:  fatal IO error 11 (Resource temporarily unavailable) on X server ":0.0"
      after 363 requests (284 known processed) with 0 events remaining.
[student@localhost ~]$ gawk -f throughput1.awk 1_out.tr
Average Throughput[kbps] = 291.78                StartTime=0.50 StopTime=9.98
[student@localhost ~]$ ns trial2.tcl
num_nodes is set 6
INITIALIZE THE LIST xListHead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
using backward compatible Agent/CBR; use Application/Traffic/CBR instead
Start of simulation..
channel.cc:sendUp - Calc highestAntennaZ_ and distCST_
highestAntennaZ_ = 1.5,  distCST_ = 550.0
SORTING LISTS ...DONE!
[student@localhost ~]$ gawk -f throughput1.awk 1_out.tr
Average Throughput[kbps] = 151.43                StartTime=0.50 StopTime=8.02
[student@localhost ~]$
```

Figure 10. Throughput and End-to-End Delay in AntTORA

10. Conclusions

The performance of the protocols TORA and Improved TORA were measured with respect to metrics like, end-to end delay, throughput and packet delivery fraction. Simulations were carried out with identical topologies and running different protocols with different number of nodes. The results of the simulation indicate that performance of TORA protocol has been improved over traditional TORA protocol. Finally the performance comparison of the two routing protocols we show that Ant colony Optimization Algorithm improves traditional TORA protocol.

REFERENCES

[1] Boukerche, Azzedine, Begumhan Turgut, Nevin Aydin, Mohammad Z. Ahmad, Ladislau Bölöni, and Damla Turgut. "Routing protocols in ad hoc networks: A survey." Elsevier Computer networks, vol.55, no.13, p. 3032-3080, 2011.

[2] Anuj K Gupta, Harsh Sadawarti, and Anil K. Verma. "MANET Routing Protocols Based on Ant Colony Optimization.", International Journal of Modeling and Optimization, vol.2, no.1, p.42, 2012.

[3] Gianni Di Caro, Frederick Ducatelle, and Luca Maria Gambardella. "AntHocNet: an ant-based hybrid routing algorithm for mobile ad hoc networks.", In International Conference on Parallel Problem Solving from Nature, Springer Berlin Heidelberg, pp. 461-470, 2004..

[4] Park Vincent Douglas, and M. Scott Corson. "A highly adaptive distributed routing algorithm for mobile wireless networks." INFOCOM'97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., Proceedings IEEE. Vol. 3, pp.1405-1413, 1997.

[5] Martins Jose Alex Pontes, Sergio Luis OB Correia, and Joaquim Celestino, "Ant-DYMO: A bio-inspired algorithm for MANETS." , Telecommunications (ICT), 2010 IEEE 17th International Conference on. IEEE, pp.748-754, 2010.